

# Predictive Current Control of Asynchronous Machines by Optimizing the Switching Moments

P. Goedertier, T.J. Vyncke, F.M. De Belie, and J.A. Melkebeek

Department of Electrical Energy, Systems and Automation (EESA)

Ghent University (UGent), Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium

phone: +32 (0)9 264 3442, fax: +32 (0)9 264 3582, e-mail: Thomas.Vyncke@UGent.be

**Abstract**—In this paper a model-based predictive control (MBPC) scheme for the current control of induction machines is presented. The controller directly selects the optimal switch state of the inverter. The proposed scheme uses a longer prediction horizon and a limited amount of optimal switching instants to reduce the average switching frequency. The next iteration of the MBPC-scheme is performed at the established optimal switching instant, as such suppressing the receding horizon property for short time spans.

The proposed method is compared to a more conventional MBPC-scheme with a very short prediction horizon. Both simulations and experiments clearly show a significant reduction in average switching frequency. However, with a reduction in switching frequency the torque ripple is increased. To correctly assess the properties of the different schemes, a key performance indicator is proposed that offers a fair and unbiased comparison in terms of switching frequency and torque ripple.

**Index Terms**—Indirect field-oriented control, induction machines, MBPC, predictive control, FPGA implementation, switching-loss reduction

## I. INTRODUCTION

Control schemes in electrical drives have been the subject of research for several decades. In practice, two typical approaches are used in industrial applications: schemes based on linear control (PI-controllers) in a field-oriented reference frame combined with a voltage modulator [1] and schemes based on non-linear control with hysteresis bounds [2].

These concepts are mainly focused on a low computational complexity. However the increasing computation power of digital hardware in the last decades has allowed the application of very demanding control schemes. Especially model-based predictive control (MBPC) [3]–[5] has received increasing interest. Within MBPC the controller uses a machine and converter model to predict the future behaviour, then the optimal control action is selected by means of a cost function. The notion of optimality for the selected control action and the natural handling of system constraints are the main reasons for the increased interest in MBPC from academia and industry.

Copyright © 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

P. Goedertier is Business Analyst at McKinsey & Company, Th. Vyncke is Consultant at VeroTech, Th. Vyncke, F. De Belie “Email: frederik.debelie@ugent.be” and J. Melkebeek “Email: jan.melkebeek@ugent.be” are with the Department of Electrical Energy, Systems and Automation of Ghent University, St-Pietersnieuwstraat 41, B-9000 Gent, Belgium.

In the MBPC-scheme, the future system states are predicted in discrete-time steps (often corresponding with the fixed sampling frequency of the controller). The number of subsequent prediction steps is called the prediction horizon  $N_y$ . For larger prediction horizons the MBPC-scheme is computationally very demanding, even for the powerful digital devices that are nowadays available.

As such two approaches exist in the literature: either the MBPC uses a long horizon (tens or hundreds of time steps) where the problem is solved offline [3] or the horizon is typically reduced to one time step (or a few sampling periods at most) and the problem is solved online [5], [6]. This problem is even more evident for complex converter topologies, such as multilevel inverters [6], [7]. The offline versions of MBPC are rather impractical, but for the short prediction horizon versions it is difficult to reduce the switching frequency properly. In this paper an online MBPC is studied and realised for medium-length prediction horizons (typically 5 to 10 periods). The experimental verification is achieved by the implementation of the prediction calculations in programmable digital hardware (FPGA), as discussed in [7]. The MBPC scheme in this paper takes the inverters discrete switch conditions directly into account and uses no modulator. As the available switch states are limited, this technique is often referred to as finite-set MBPC (FS-MBPC).

The literature on the model-based predictive control of AC machines is mainly focused on the direct control of torque and flux by choosing the best inverter switch state (often called predictive torque control, PTC) [3], [8]–[10], a technique closely related to the more conventional DTC-technique (direct torque control [2]). Considerably less publications discuss the direct current control [4], [11]–[13] of AC machines. The control task here is to accurately track the reference current obtained from a field-oriented control principle. The proposed method reduces the switching frequency efficiently without compromising the current tracking (and thus the torque quality) too much.

## II. FIELD-ORIENTED CONTROL OF INDUCTION MACHINES

The field-oriented control of induction machines [14], [15], has become a standard technique in industry. The main principles are summarized below.

### A. Modelling Induction Machines

Induction machines can be modelled by an equivalent two-phase speed-dependent fourth-order state-space equation. The

horizontal and vertical axis of the stationary reference frame are called  $\alpha$  and  $\beta$ , respectively. The measured stator current  $i_s$  and unmeasured rotor flux  $\psi_r$  are chosen as overall state vector  $\mathbf{x} = [i_{s\alpha} \ i_{s\beta} \ \psi_{r\alpha} \ \psi_{r\beta}]^T$ . They are controlled by the stator voltage  $\mathbf{v}_s$ .

In the following equations  $N_p$  denotes the pole pairs.  $L_s$ ,  $R_s$ ,  $R_r$  and  $\omega_r$  are respectively the stator inductance, stator resistance, rotor resistance and rotor speed. Assuming no leakage on rotor side,  $L_m$  is both mutual and rotor inductance whereas the stator leakage inductance  $L_{s\sigma}$  is  $L_s - L_m$ .

The system dynamics are described by the state-space model

$$\dot{\mathbf{x}} = \mathbf{A}(\omega_r)\mathbf{x} + \mathbf{B}\mathbf{v}_s \quad (1)$$

$$\mathbf{v}_s = [v_{s\alpha} \ v_{s\beta}]^T \quad (2)$$

$$\mathbf{A}(\omega_r) = \begin{bmatrix} -\frac{R_s+R_r}{L_{s\sigma}} & 0 & \frac{R_r}{L_{s\sigma}L_m} & \frac{\omega_r}{L_{s\sigma}} \\ 0 & -\frac{R_s+R_r}{L_{s\sigma}} & -\frac{\omega_r}{L_{s\sigma}} & \frac{R_r}{L_{s\sigma}L_m} \\ R_r & 0 & -\frac{R_r}{L_m} & -\omega_r \\ 0 & R_r & \omega_r & -\frac{R_r}{L_m} \end{bmatrix} \quad (3)$$

$$\mathbf{B} = \begin{bmatrix} \frac{1}{L_{s\sigma}} & 0 \\ 0 & \frac{1}{L_{s\sigma}} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \quad (4)$$

The electromagnetic torque  $t_{em}$  is given by

$$t_{em} = \frac{3}{2}N_pL_m(\psi_{r\beta}i_{s\alpha} - \psi_{r\alpha}i_{s\beta}). \quad (5)$$

### B. Indirect Field-Oriented Control - Current References

Using a reference frame (d,q), with the d-axis fixed to the rotor-flux vector and angular speed  $\omega_{dq} = \omega_{slip} + \omega_r$  relative to the stationary frame ( $\alpha, \beta$ ), the stator current can be decomposed in two perpendicular components:  $i_{sd}$  generating the rotor flux  $\psi_{rd}$  and a component  $i_{sq}$  responsible for the torque  $t_{em}$ . The dynamics of this technique illustrated on Fig. 1 and called ‘indirect field-oriented control’ [14], [15] are:

$$\psi_{rd}(p) = \frac{L_m}{1 + p\frac{L_m}{R_r}}i_{sd}(p) \quad (6)$$

$$t_{em}(t) = \frac{3}{2}N_pL_m\psi_{rd}(t)i_{sq}(t) \quad (7)$$

$$\omega_{slip}(t) = -\frac{R_r i_{sq}(t)}{\psi_{rd}(t)}; \quad \theta_{slip}(t) = \int_{t_0}^t \omega_{slip}(t')dt'. \quad (8)$$

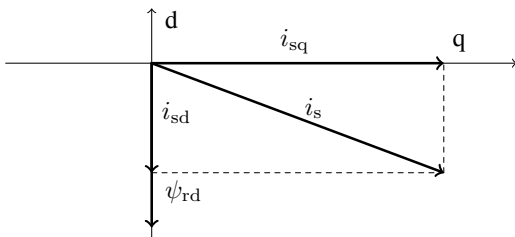


Fig. 1: dq-components of the stator current and rotor flux

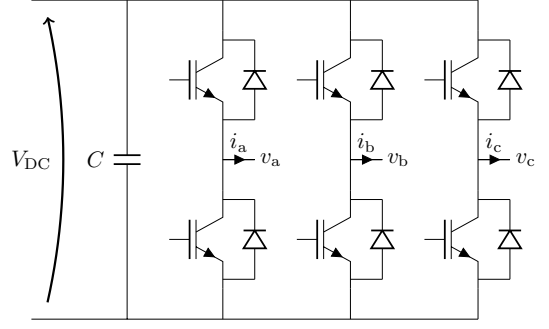


Fig. 2: 2-level voltage-source inverter

Table I: The 8 voltage vectors depending on the inverter state

Binary code	$v_a$	$v_b$	$v_c$	$v_\alpha$	$v_\beta$
000	$-\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	0	0
001	$-\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	$-\frac{1}{3}V_{DC}$	$-\frac{\sqrt{3}}{3}V_{DC}$
010	$-\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	$-\frac{1}{3}V_{DC}$	$\frac{\sqrt{3}}{3}V_{DC}$
011	$-\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	$-\frac{2}{3}V_{DC}$	0
100	$\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	$\frac{2}{3}V_{DC}$	0
101	$\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	$\frac{1}{3}V_{DC}$	$-\frac{\sqrt{3}}{3}V_{DC}$
110	$\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	$-\frac{V_{DC}}{2}$	$\frac{1}{3}V_{DC}$	$\frac{\sqrt{3}}{3}V_{DC}$
111	$\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	$\frac{V_{DC}}{2}$	0	0

$p$  is the Laplace-operator. Reference values  $\psi_{rd}^{ref}$  and  $t_{em}^{ref}$  can be transformed into references  $i_{sq}^{ref}$  and  $i_{sd}^{ref}$ , which can be transformed into reference values  $i_{s\alpha}^{ref}$  and  $i_{s\beta}^{ref}$ :

$$\begin{bmatrix} i_{s\alpha}^{ref} \\ i_{s\beta}^{ref} \end{bmatrix} = \begin{bmatrix} \cos(\theta_{dq}) & -\sin(\theta_{dq}) \\ \sin(\theta_{dq}) & \cos(\theta_{dq}) \end{bmatrix} \begin{bmatrix} i_{sq}^{ref} \\ i_{sd}^{ref} \end{bmatrix}; \quad \theta_{dq} = \theta_{slip} + \theta_r \quad (9)$$

In this paper the challenge is to track the reference current through a voltage-source inverter with minimal switching-losses. This paper deals with a two-level inverter as shown in Fig. 2, but the principles can be extended to other inverter topologies. The three-phase currents and the dc-voltage are measured. Table I contains a list of the 8 instantaneous stator voltage vectors. The binary code indicates the state of the semiconductor switches: the MSB stands for phase a, the LSB is phase c and 1 or 0 indicates the on-state of respectively the upper or lower switch. Mind that there is no difference between 000 and 111 except for the homopolar component generating zero rotor flux nor electromagnetic torque.

### III. PREDICTIVE CURRENT CONTROL WITH OPTIMIZATION OF THE VOLTAGE VECTORS AND SWITCHING MOMENTS

The reference current obtained with the field-oriented control principle has to be realized by the predictive current controller, as shown in figure 3.

#### A. FS-MBPC in General

The FS-MBPC algorithm has three important steps [5], [6]:

- 1) ESTIMATION: based on samples of currents, dc-voltage, rotor speed and rotor position at update instant  $k-1$  an

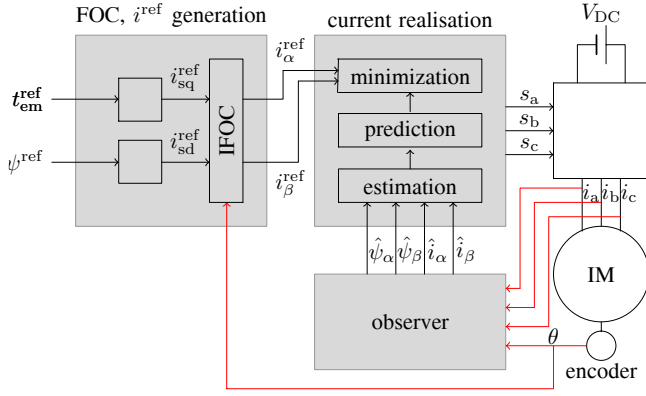


Fig. 3: Overall control scheme

estimation is made for the state at update instant  $k$ . Up to that moment  $k$ , the switch state applied at  $k - 1$  is considered.

- 2) PREDICTION: starting from the estimated state at  $k$ , predictions are made for the state evolution over the entire prediction horizon  $N_y$  (from  $k$  to  $k + N_y$ ) for each allowed sequence of switch states.
- 3) OPTIMISATION: for each switch state sequence a cost function for the states from  $k + 1$  to  $k + N_y$  is calculated. The sequence  $k + 1$  to  $k + N_y$  with minimal cost is then selected and the first switch state is applied. After this, the algorithm is iterated.

The sampling period for the algorithm and thus the update period of the switch state is denoted by  $t_\Delta$ . The update frequency  $f_u = \frac{1}{t_\Delta}$  is much higher than the average switching frequency. Still, the average switching frequency can be rather high. As the prediction horizon often is restricted to  $N_y = 1$  to realize online implementation, switching frequency reduction can not satisfactorily be obtained.

For the estimation and prediction steps the model in (3) is used. However, the rotor flux is not measured and as such an observer based on the measured states (current, voltage and position) is used as shown in Fig. 3.

### B. State Estimation: Observer for the Rotor Flux

A hands-on approach is to take the estimated stator current  $\hat{i}_s$  equal to the measured one  $\tilde{i}_s$ . The rotor flux is estimated by using (3).

$$\begin{aligned} \hat{i}_{s\alpha}(n+1) &= \tilde{i}_{s\alpha}(n+1) \\ \hat{i}_{s\beta}(n+1) &= \tilde{i}_{s\beta}(n+1) \\ \hat{\psi}_{s\alpha}(n+1) &= t_\Delta R_r \hat{i}_{s\alpha}(n) + \left[1 - \frac{t_\Delta R_r}{L_m}\right] \hat{\psi}_{s\alpha}(n) - t_\Delta \omega_r \hat{\psi}_{s\beta}(n) \\ \hat{\psi}_{s\beta}(n+1) &= t_\Delta R_r \hat{i}_{s\beta}(n) + \left[1 - \frac{t_\Delta R_r}{L_m}\right] \hat{\psi}_{s\beta}(n) + t_\Delta \omega_r \hat{\psi}_{s\alpha}(n) \end{aligned} \quad (10)$$

Both simulations and experiments show a good robustness of the current control against machine parameter variations, even with this very simple state estimator.

### C. New Algorithm: Long Horizon, Few Switches

The newly proposed algorithm searches not only for the optimal voltage vectors, but also for the best switching moments. In this paper, two switching moments are allowed over

the total horizon only: one in the beginning and one on an arbitrary point of time within the prediction horizon. So the prediction horizon is divided into two time intervals each with their own optimal voltage vector.

The first optimal voltage vector  $v_{ap}$  (ap: apply) will be applied during its respective optimal time whereas the second voltage vector  $v_f$  (f: foresee) has to make sure the control system does not fall into a situation where it will be difficult to track the reference current in the future. This method is called the Long Horizon Few Switches (LHFS) algorithm. The main difference between the LHFS-algorithm and other methods proposed in literature with a longer horizon [3], [13] is twofold:

- the horizon is still less than 10 update periods (compared to several tens or hundreds of [3], [13]),
- the fact that the receding horizon property is suppressed during  $m_{opt}$  sample periods. This results in a smoother control without the need for an explicit ‘end-cost’.

The LHFS algorithm consists of the following steps:

- 1) construct all possible sequences of voltage vectors

$$\underbrace{[v_{ap} \dots v_{ap}]_{N_y - m}}_{N_y - m} \underbrace{[v_f \dots v_f]_m}_m; \quad 0 \leq m < N_y \quad (11)$$

$$v_{ap} \text{ and } v_f \in [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6]; \quad v_{ap} \neq v_f \quad (12)$$

Only one null vector is considered.  $v_{ap}$  and  $v_f$  have respectively 7 and 6 possibilities.

- 2) Simulate for all sequences the current evolution and associate a cost with it.

$$K = \sum_{j=t_\Delta}^{N_y t_\Delta} (i_{sq}(t+j) - i_{sq}^{ref})^2 + (i_{sd}(t+j) - i_{sd}^{ref})^2 \quad (13)$$

- 3) Determine the optimal sequence, i.e. the one with lowest associated cost.
- 4) Apply the optimal vector  $v_{ap,opt}$  during  $N_y - m_{opt}$  samples and repeat the whole calculation. If  $v_{a,opt}$  is a null vector, then switch to the null vector that requires least switches (according to Table I).

The above algorithm is rather hard to evaluate within an update period for long horizons and/or other inverter topologies with more than two voltage levels. In a naive implementation where all sequences are simulated after each other,

$$42N_y^2; \quad N_y > 1; \quad \mathcal{O}(N_y^2) \quad (14)$$

time steps have to be simulated since there are 7 possibilities for  $v_{ap}$ , 6 for  $v_f$  and  $N_y$  for  $m$ . Finally every combination of  $v_{ap}$ ,  $v_f$  and  $m$  has to be simulated over  $N_y$  time steps. As e.g. the sequence  $[1 \ 1 \ 1 \ 5 \ 5]$  and  $[1 \ 1 \ 6 \ 6 \ 6]$  both start with  $[1 \ 1]$ , this common switching sequence can be reused.

In general the voltage vector of the common sequence has 7 possibilities and needs to be simulated over  $N_y$  time steps. As such,  $v_f$  has 6 possibilities and needs to be simulated over 1 to  $N_y - 1$  time steps. Hence the number of calculations is

$$7 \left( N_y + 6 \sum_{j=1}^{N_y-1} j \right) = 21N_y^2 - 14N_y; \quad \mathcal{O}(N_y^2). \quad (15)$$

Compared to (14), this is a reduction with more than a half, but it complicates the implementation.

#### D. Simplification

A further reduction of the complexity of the algorithm can be made using the so-called ‘Branch & Bound’-techniques. This means that some voltage sequences are considered inappropriate in advance, so their current evolutions don’t have to be simulated, nor evaluated.

Specifically,  $v_{ap}$  may *at maximum* differ one single switch of a random phase from the previous applied voltage vector<sup>1</sup>, subsequently  $v_f$  *must* differ one switch from its respective  $v_{ap}$ . As the amount of switches is important, it is clear that the two null vectors have to be taken in account directly. Hence  $v_{ap}$  has 4 possibilities,  $v_f$  3 and  $m$  is still going from 1 to  $N_y - 1$ . The total calculations amount to

$$12N_y^2; \quad \mathcal{O}(N_y^2) \quad (16)$$

for a naive implementation, for the reusing one it is

$$4 \left( N_y + 3 \sum_{j=1}^{N_y-1} j \right) = 6N_y^2 - 2N_y; \quad \mathcal{O}(N_y^2). \quad (17)$$

For prediction horizons up to  $N_y = 20$  the number of calculations is comparable to the cases discussed in [7], where the FPGA-implementation of FS-MBPC is discussed.

#### IV. KEY PERFORMANCE INDICATOR

The first goal of the controller is to track the reference current precisely to minimize the torque ripple.

The expected electromagnetic torque strongly depends on the rotor inductance and resistance as can be seen in eq. (6) to (8). Biased machine parameters will give biased fluxes and torques, even with a perfect current controller. This systematic error during experiments due to the error on the accepted, yet slightly inaccurate, machine parameters is unimportant and therefore it’s undesirable to take it into account as torque ripple. Consequently, the root mean square error (RMSE) of the torque compared to the average measured torque is chosen as criterion for the torque ripple

$$\text{RMSE}_{t_{em}} = \sqrt{\frac{\sum_{n=1}^{n_{total}} (t_{em}(n) - \bar{t}_{em}(n))^2}{n_{total}}} \quad (18)$$

The second goal is to minimize the average switching frequency of an inverter leg  $f_{switch}$ .

As the algorithm prohibits the controller to switch every update period it is clear that the average switching frequency per leg of the inverter  $f_{switch}$  will lower with longer horizons. The torque ripple on the other side will rise. Therefore it’s not

<sup>1</sup>This means the binary code (table I) of  $v_{ap}$  and the actual applied voltage vector may differ one random bit

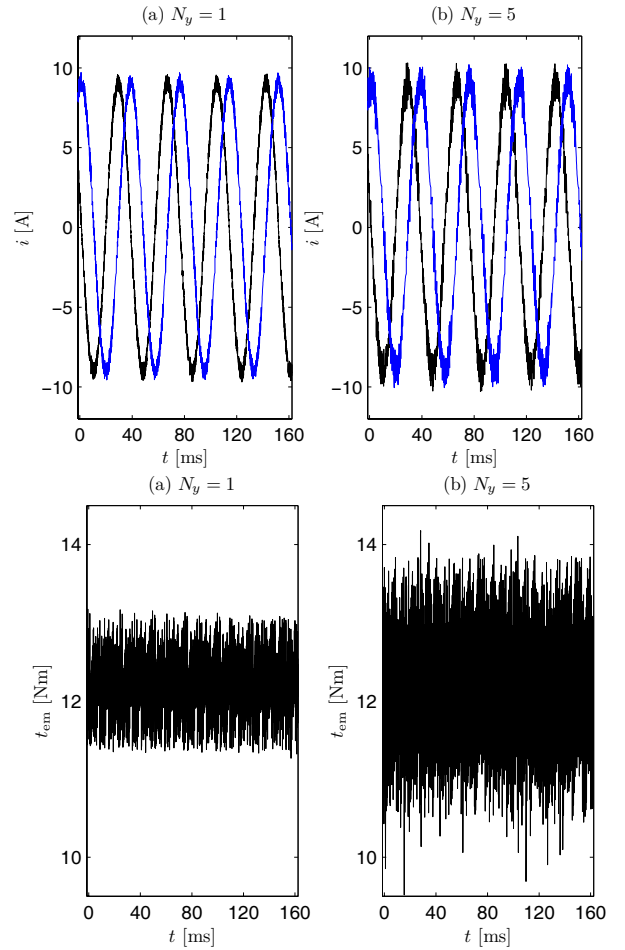


Fig. 4: Simulated current control with a one-step-ahead algorithm ( $N_y = 1$ ) and the LHFS ( $N_y = 5$ ) algorithm. Above:  $\alpha$  and  $\beta$  currents, below: torque. Simulation conditions:  $i_{sq} = 8.5\text{A}$ ,  $i_{sd} = -3.2\text{A}$ ,  $\frac{\omega_r}{\omega_{s, \text{nom}}} = 0.5$  and  $V_{DC} = 538\text{V}$

directly clear whether this method is to be preferred above an ordinary one-step-ahead prediction with a lower update frequency since it also has lower switching frequency and higher torque ripple.

A good comparison takes this balance in account. The product  $f_{switch} \text{RMSE}_{t_{em}}$  appears to be characterizing for a certain control strategy and working point, independent from  $f_{update}$ <sup>2</sup>, as will be shown in the next section. Therefore it is taken as the key performance indicator (KPI).

Good control strategies have lower KPI’s compared to worse strategies as their torque ripple is lower for the same switching strategy. Alternatively their switching frequency is lower for the same torque ripple.

#### V. SIMULATIONS

The simulations and experiments are done on an induction machine with parameters  $N_p = 1$ ,  $f_{s, \text{nom}} = 50\text{Hz}$ ,  $R_s = 1.26\Omega$ ,  $L_s = 0.304\text{H}$ ,  $R_r = 1\Omega$  and  $L_r = L_m = 0.28\text{H}$ .

In Fig. 4 the current control and obtained torque is compared for the algorithm with  $N_y = 1$  (one-step-ahead) and LHFS

<sup>2</sup>As long as  $f_{switch} > 2\text{kHz}$  for the in the next section considered machine

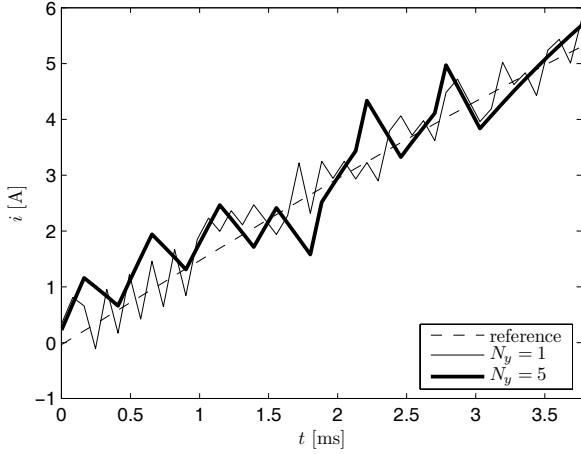


Fig. 5: Current ripple with a one-step-ahead algorithm ( $N_y = 1$ ) and the LHFS ( $N_y = 5$ ) algorithm (zoomed view).

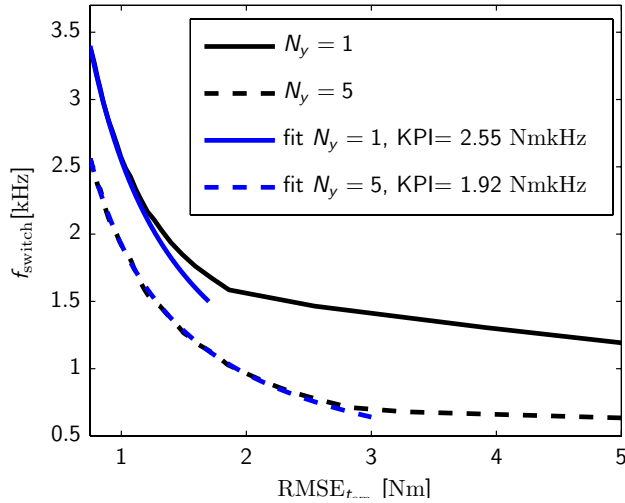


Fig. 6: At high enough  $f_{\text{update}}$ , the product  $f_{\text{switch}} \text{RMSE}_{t_{\text{em}}}$  is constant. For low  $f_{\text{update}}$ , the KPI rises. Simulation conditions:  $i_{\text{sq}} = 8.5\text{A}$ ,  $i_{\text{sd}} = -3.2\text{A}$ ,  $\frac{\omega_r}{\omega_{\text{sy,nom}}} = 0.5$  and  $V_{\text{DC}} = 538\text{V}$

with  $N_y = 5$ . Clearly both algorithms are capable of tracking the current reference, but the LHFS algorithm has a higher torque ripple. In Fig. 5 however it is clear that also the switching frequency is lower. A proper comparison requires the use of the KPI.

#### A. Original Algorithm

Fig. 6 shows the torque ripple in function of the switching frequency. The update frequency is the implicit parameter.

It follows that the proposed algorithm ( $N_y = 5$ ) performs better than a classic one-step-ahead predictor ( $N_y = 1$ ) as the same torque ripple can be achieved with fewer switches. Moreover the simulated curves can be fit by  $f_{\text{switch}} \text{RMSE}_{t_{\text{em}}} = \text{constant}$ . This proves that the KPI is independent of  $f_{\text{update}}$  for medium switching frequencies and higher ( $> 2\text{kHz}$ ). At low  $f_{\text{switch}} \approx 1\text{kHz}$ , the KPI increases since the torque ripple still rises with declining  $f_{\text{update}}$ , but  $f_{\text{switch}}$  is rather constant. Nev-

ertheless the proposed algorithm is an improvement relative to a one-step-ahead prediction.

Fig. 7 presents the influence of the speed. The properties for  $N_y = 5$  (and  $N_y = 3$ ), compared to a classic  $N_y = 1$  predictor, are:

- (a) From  $\omega_r > 0.1$  ( $0.25$ )  $\omega_{\text{sy,nom}}$ , the torque ripple rises.
- (b) Simultaneously  $f_{\text{switch}}$  saturates, which means it is relatively constant over a large speed range.
- (c) The KPI rises with increasing speed, but not as much as for  $N_y = 1$ .
- (d) Especially at medium speeds ( $\frac{\omega_r}{\omega_{\text{sy,nom}}} \approx 0.5$ ), there is a remarkable improvement: the KPI is 25% (20%) lower.

#### B. Simplified Algorithm

Fig. 8 indicates that the simplified algorithm also has a constant KPI for not too low update frequencies. Moreover, the forced reduction of switching possibilities has a good impact on the overall performance as the curves of the simplified

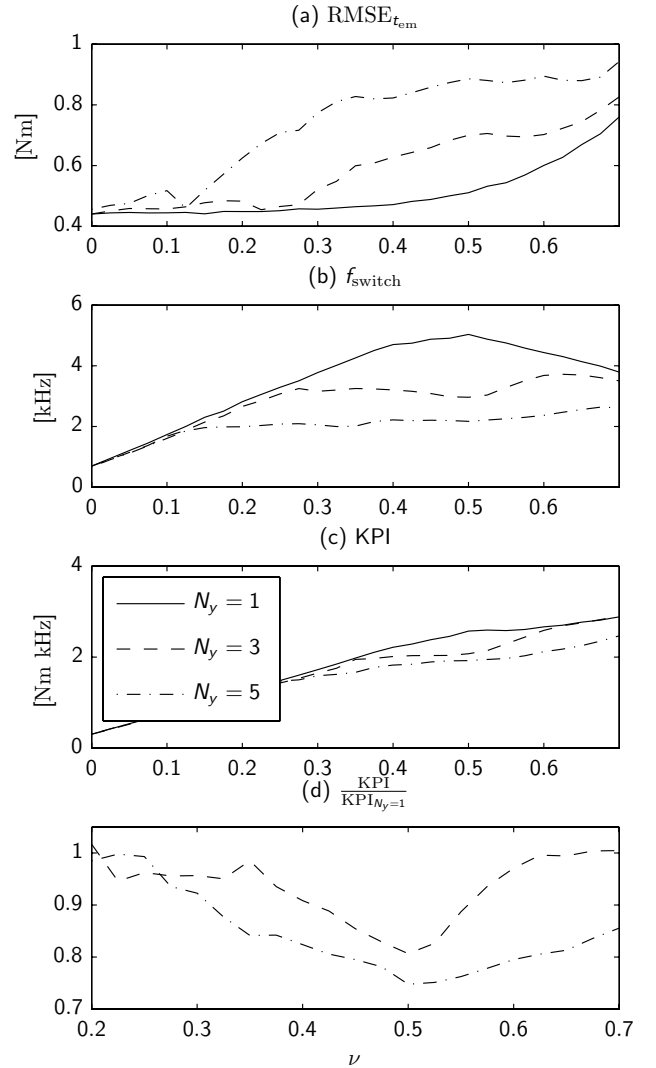


Fig. 7: Influence of the speed on the performance of the algorithm.

Enlarging the horizon increases torque ripple (a), but this is overcompensated by the lower switching frequency (b) as seen in (c) and (d). Simulation conditions:  $i_{\text{sq}} = 8.5\text{A}$ ,  $i_{\text{sd}} = -3.2\text{A}$ ,  $f_{\text{update}} = 12.2\text{kHz}$  and  $V_{\text{DC}} = 538\text{V}$

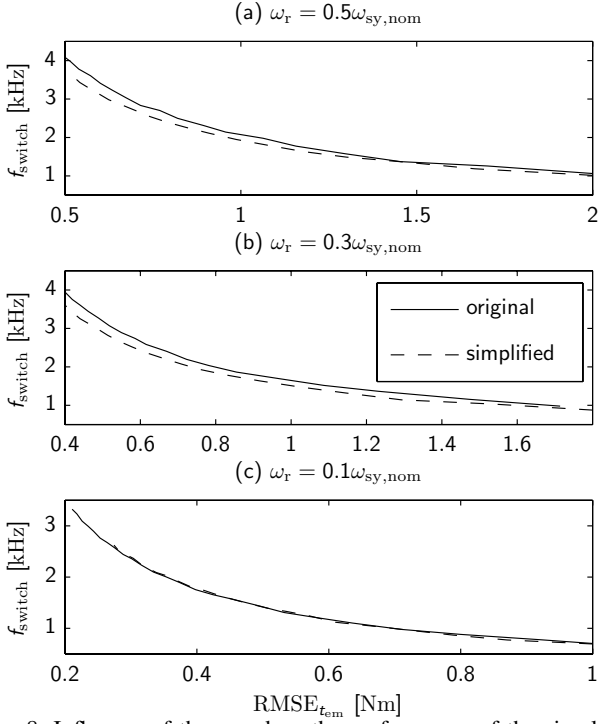


Fig. 8: Influence of the speed on the performance of the simplified algorithm. It performs at least as good as the original algorithm. Simulation conditions:  $i_{sq} = 5\text{A}$ ,  $i_{sd} = -3.2\text{A}$ ,  $N_y = 5$  and  $V_{DC} = 538\text{V}$

algorithm are situated lower than those of the original. This behaviour is comfortable since the simplified algorithm is both easier to implement while it still performs better.

## VI. EXPERIMENTS

To verify the results, the *simplified* algorithm was used in the experiments. The control was implemented in a Xilinx Spartan 6 evaluation board (Atlys) from Digilent Inc. with the System Generator Toolbox for Matlab/Simulink. The Chipscope tool was used to obtain measurements. The 2-level inverter is realized with a Fuji Electronic. IPM (Intelligent Power Module) rated at 600V, 15A. The update frequency is 12 kHz.

Fig. 9 shows the current evolution as a function of time for a one step ahead predictor while Fig. 10 shows the same working point for the proposed algorithm. As it is impossible to say at sight which one is the better, calculation of the KPI is required.

In Table II the switching frequency, estimated torque ripple and KPI are given for a prediction horizon  $N_y$  of respectively 1 and 5. The torque ripple is estimated through the current ripple and the model since dynamic torques (especially high-frequency components) are difficult to measure. The average switching frequency is obtained by monitoring the voltage vector command over a long time range.

The experimental results confirm the results obtained in simulation. A longer horizon, and thus less abilities to switch the voltage vector, obviously lowers the switching frequency, but it also increases the torque ripple. Clearly the algorithm with  $N_y = 5$  is preferable: although the torque ripple almost doubles, the switching frequency is reduced with almost a factor 3.

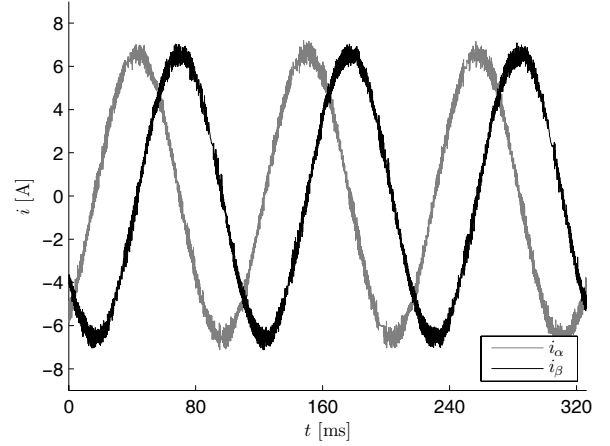


Fig. 9: Current as a function of time for a one step ahead predictor. Experiment conditions:  $i_{sq} = 6\text{A}$ ,  $i_{sd} = -3\text{A}$ ,  $\omega_r = 500\text{rpm}$  and  $V_{DC} = 150\text{V}$

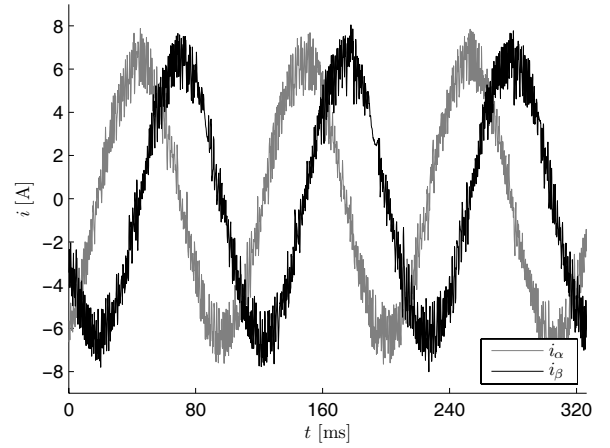


Fig. 10: Current as a function of time for the proposed algorithm with  $N_y = 5$ . Experiment conditions:  $i_{sq} = 6\text{A}$ ,  $i_{sd} = -3\text{A}$ ,  $\omega_r = 500\text{rpm}$  and  $V_{DC} = 150\text{V}$

As mentioned before, the KPI takes both effects into account. For the longer horizon ( $N_y = 5$ ) this results in a KPI which is 27% lower compared to the one-step-ahead solution. The LHFS-algorithm clearly is capable of closely tracking the reference current with a very low effective switching frequency whilst limiting the torque ripple.

## VII. CONCLUSIONS

In this paper a model-based predictive controller was presented with a medium-length prediction horizon and optimized switching instants. This algorithm allows the reduction of the average switching frequency whilst obtaining good current tracking and thus high-quality torque control. This was verified

$N_y$	1	5
KPI [Nm kHz]	1.059	0.771
$\hat{\sigma}_{t_{em}}$ [Nm]	0.300	0.596
$f_{switch}$ [kHz]	3.529	1.294

Table II: Experimental conditions:  $i_{sq} = 6\text{A}$ ,  $i_{sd} = -3\text{A}$ ,  $\omega_r = 500\text{rpm}$  and  $V_{DC} = 150\text{V}$

by simulation as well as experiment. A Key Performance Indicator was proposed to correctly compare several algorithms.

#### ACKNOWLEDGMENT

This work was supported by the Interuniversity Attraction Pole under Project (IUAP) P7/02. The work of T.J. Vyncke was supported by a Ph.D. fellowship from the Research Foundation Flanders (FWO).

#### REFERENCES

- [1] F. Blaschke, "The principle of field orientation as applied to the new transvector closed-loop control system for rotating field machines," *Siemens Review*, vol. 39, no. 5, pp. 217–219, 1972.
- [2] I. Takahashi and T. Noguchi, "A new quick-response and high-efficiency control strategy of an induction motor," *IEEE Trans. Ind. Appl.*, vol. 22, no. 5, pp. 820–827, Sep./Oct. 1986.
- [3] T. Geyer, "Low complexity model predictive control in power electronics and power systems," Ph.D. dissertation, Swiss Federal Institute of Technology (ETH) Zürich, 2005.
- [4] A. Linder, R. Kanchan, R. Kennel, and P. Stolze, *Model-Based predictive control of electric drives*. Göttingen: Cuvillier Verlag, 2010.
- [5] S. Kouro, P. Cortés, R. Vargas, U. Ammann, and J. Rodríguez, "Model predictive control—a simple and powerful method to control power converters," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1826–1838, Jun. 2009.
- [6] T. J. Vyncke, S. Thielemans, M. Jaxsens, and J. A. A. Melkebeek, "Analysis of design choices in model based predictive control of flying-capacitor inverters," *COMPEL-The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 31, no. 2, pp. 619–635, Feb. 2012.
- [7] T. J. Vyncke, S. Thielemans, and J. A. Melkebeek, "Finite-set model based predictive control for flying-capacitor inverters: cost function design and efficient FPGA implementation," *IEEE Tran. Ind. Informatics*, doi 10.1109/TII.2012.2223707 accepted for publication.
- [8] S. Bolognani, S. Bolognani, L. Peretti, and M. Zigliotto, "Design and implementation of model predictive control for electrical motor drives," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1925–1936, Jun. 2009.
- [9] H. Miranda, P. Cortés, J. I. Yuz, and J. Rodríguez, "Predictive torque control of induction machines based on state-space models," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1916–1924, Jun. 2009.
- [10] M. Preindl and S. Bolognani, "Model predictive direct speed control with finite control set of PMSM-VSI drive systems," in *Workshop on Predictive Control of Electrical Drives and Power Electronics (PRECEDE)*, Munich, Germany, Oct. 14–15 2011, pp. 17–23.
- [11] M. Preindl, E. Schaltz, and P. Thøgersen, "Switching frequency reduction using model predictive direct current control for high-power voltage source inverters," *IEEE Trans. Ind. Electron.*, vol. 58, no. 7, pp. 2826–2835, Jul. 2011.
- [12] T. Geyer, N. Oikonomou, G. Papafotiou, and F. D. Kieferndorf, "Model predictive pulse pattern control," *IEEE Trans. Ind. Appl.*, vol. 48, no. 2, pp. 663–676, March–April 2012.
- [13] T. Geyer, "Model predictive direct current control: Formulation of the stator current bounds and the concept of the switching horizon," *IEEE Ind. Appl. Mag.*, vol. 18, no. 2, pp. 47–59, March–April 2012.
- [14] R. W. De Doncker and D. W. Novotny, "The universal field oriented controller," *IEEE Trans. Ind. Appl.*, vol. 30, no. 1, Jan./Feb. 1994.
- [15] D. Novotny and T. Lipo, *Vector Control and Dynamics of AC Drives*. London: Oxford University Press, 1996.